Research of Genetic Training Algorithm for Identifying Mechanical Failure Modes within the Framework of Case-Based Reasoning

XU Yuan-ming, ZHANG Yang, CHEN Li-na

(School of Aeronautical Science and Engineering Technology, Beijing University of Aeronautics and Astronautics, Beijing 100083, China)

Abstract: The combination of case-based reasoning (CBR) and genetic algorithm (GA) is considered in the problem of failure mode identification in aeronautical component failure analysis. Several implementation issues such as matching attributes selection, similarity measure calculation, weights learning and training evaluation policies are carefully studied. The testing applications illustrate that an accuracy of 74.67% can be achieved with 75 balanced-distributed failure cases covering 3 failure modes, and that the resulting learning weight vector can be well applied to the other 2 failure modes, achieving 73.3% of recognition accuracy. It is also proved that its popularizing capability is good to the recognition of even more mixed failure modes.

Key words: failure mode identification; case-based reasoning; genetic algorithm; learning train

实例推理中遗传训练算法用于机械失效模式识别的研究. 徐元铭, 张洋, 陈丽娜. 中国航空学报(英文版), 2005, 18(2): 122-129.

摘 要: 采用实例推理和遗传算法相结合的方法, 研究了航空机械零部件失效模式识别的问题。对用于识别的失效属性的选择、检索相似度计算、训练用遗传算法的适应度函数设计以及训练策略的影响进行了较为详细的描述。应用测试表明, 对包含分布均衡的 3 种模式的情况取得了高于 74.67% 的识别率, 所获得的最佳权值向量对另外 2 种模式具有很好的识别精度(大于 73.3%), 对混合多模式情况也具有较好的推广能力。验证了该方法对航空零部件失效模式的识别是可行的。

关键词: 失效模式识别; 基于实例的推理; 遗传算法; 学习训练

文章编号: 1000-9361(2005)02-0122-08 中图分类号: TP391.5; TP311.11 文献标识码: A

1 Introduction

Failure modes normally refer to those forms of exhibiting a failure of a mechanical component in either macroscopic or microscopic sense, or the classification of the component's failure mechanisms according to the physical, chemical or other processes which have led to a failure. Identifying failure mode of a failed component is the most important step in the entire task of failure analysis, since it can give the effective task-oriented guidelines for subsequent analysis decisions on determining failure causes and recommending precaution actions.

In aeronautical equipment failure analysis domain, failure mode identification is usually complex and time consuming. It involves, in many cases, a group of experts for making synthetic decisions[1]. The application of logic based Artificial Intelligence (AI) techniques gives a promising method for aiding human's failure analysis task. Several noticeable research work have been done in this research area, but still very limited: Mayer[2] used an expert system approach to identify the basic boiler tube failure mechanisms; Komai, *et al*[3] investigated image process and pattern recognition for identifying six different fracture surface morphologies; Liao, *et al*[4, 5] integrated database with expert systems, as well as case based reasoning for failure mechanism recognition in petro-chemical industry application; and Xu, *et al*[6,7], gave an rule

Received date: 2003-11-17; Revision received date: 2005-01-25
Foundation item: 973 Foundation Program of China (G1999065010)

based uncertainty expert system for aeronautical equipment failure analysis, and investigated case based reasoning (CBR) in this domain.

This paper describes further the study results of Genetic Algorithm (GA) based training with CBR problem-solving paradigms for failure mode identification of aeronautical equipment.

2 Failure Mode Identification Using CBR

CBR is a methodology which stems from human's reasoning behavior by recalling or resembling past similar situations. Its basic idea is treating first the problem to be solved as "Target Case", and a group of already solved or old problems as "Base Cases"; and proceeding with the assessment of a similarity between the target case and the base cases by designing proper quantitative "weights of judgment" schemes. Based on the degree of similarity, the solution pattern (or solution itself) of an old problem can be processed or adapted to infer the new solution pattern (or even solution) for the target problem. The advantages of applying CBR in failure analysis can be summarized as

(1) It does not require the explicit domain knowledge information, only a collection of failure cases needed to be stored, thus avoiding the bottleneck of knowledge elicitation.

(2) The mature and advanced database technology can be used to manage these failure cases.

(3) The identification capability can be incrementally improved by learning through new cases.

Fig. 1 shows the flow chart of CBR system for failure mode identification. The whole system can be characterized by two stages: case retrieval and weight learning. Case retrieval starts by inputting sufficient information about target case to be considered, and uses the attribute selection criteria (see Section 3 below) to extract the most significant relevant failure attributes associated with the identification process. The match between the selected attributes of the target case and those in case base is conducted by a similarity measure using a weighed K Nearest Neighbors (KNN) technique (see Section 4). The robust optimal weight of an

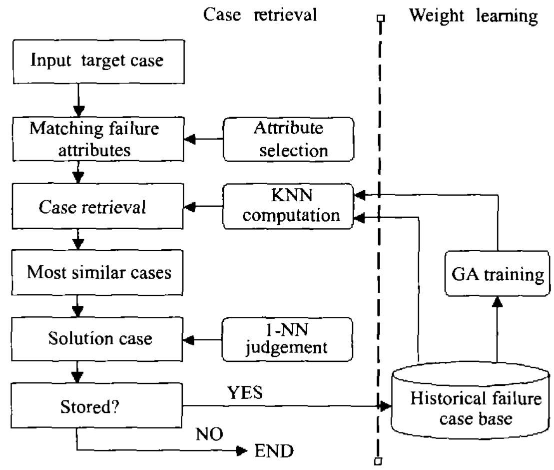


Fig. 1 The flow chart of CBR system

attribute or an attribute vector is determined by weight learning process through GA based training (see Section 5) upon a historical failure case base. Finally, the system outputs the most similar failure mode which can be validated by user to decide whether it is to be stored into the case base or not.

3 Failure Attribute Selection and Grouping

It is common sense in aeronautical equipment failure analysis that the aspects of general visual or surface conditions of a failed component, its fracture/crack face features, cross-section/subsurface features, and electron fractographic features, *etc.* have to be examined before the detailed analyses of material composition and physical properties (*e.g.*, hardness, brittle, ductility, *etc*). Failure mode identification phase requires the consideration of those features which are universal and closely related to failure occurrence and evolution of the failed component; whereas the basic information about the component's normal states, working conditions, and external loads, along with the analyses of material composition and mechanical properties and so on, are not obligatory (practices demonstrate that they are in fact more useful for failure cause determination). Therefore, the classification of failure attributes selected for failure mode identification (after consulting human experts and failure analysis handbooks) can be shown in Fig. 2.

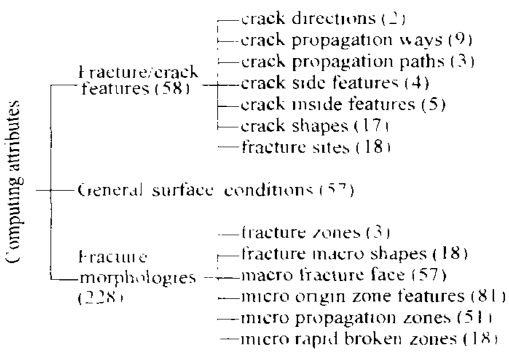


Fig. 2 The hierarchy of failure attributes

Three basic attribute groups, namely, general surface conditions (57), fracture/crack features (58) and fracture morphologies (228), are defined in this research. The number attached here represents the total number of elements that each group can take. Each group can be further divided into one or more attribute vectors as shown in Fig. 2. It should be noted that different ways of grouping failure attributes are possible. It really depends on their effects on the GA based weight training computation cost and identification accuracy.

4 A Weighed KNN Retrieval Method

The general form of similarity measure function is as follows:

$$\mathrm{SIM}(X, Y) = \frac{\sum_{i=1}^{n} W_i \times (1 - \mathrm{dist}(x_i, y_i))}{\sum_{i=1}^{n} W_i}$$

where SIM(X, Y) is the similarity between case X and Y, W_i is the weight of attribute element i, n is the number of attributes; dist(x_i, y_i) is the normalized distance of the ith attribute between two cases, and takes the form as follows:

$$\mathrm{dist}(x_i, y_i) = | x_i - y_i | / | \max(i) - \min(i) |$$

where x_i, y_i are ith attribute values of case X and Y respectively, max(i) and min(i) denote the upper and lower values of the ith attribute respectively. If x_i or y_i is unknown, which means just one of the cases or both has missing attribute values, it usually sets dist(x_i, y_i) = 0.5[5].

The assignment of the attribute values for similarity measures uses Boolean logic: 1 − existence of the attribute; − 1− no existence of the attribute; 0− unknown of the attribute.

5 Weight Learning by GA Based Training

The purpose of using Genetic Algorithm in CBR is to acquire the robust optimal weights of failure attributes upon an original failure case base. This is an important part of failure mode identification since it has a great influence on identification performance. Because the total numbers of attribute elements and attribute vectors are 343 and 14 (see Fig. 2) respectively, it would be unrealistic to assign a weight for each attribute element or each attribute vector due to the computation burden. Therefore, it is decided to set a same weight for attribute elements or vectors in the same group. In this way, The total number of weights used for training is 3, *i. e.* weight vector $W = [w_1, w_2, w_3]$. The weight assignment for case retrieval is performed based on the searching and learning capabilites of GA. GA is the optimization algorithm based on the natural evolution concept coming from Darwin's theory of evolution. The natural selection increases the surviving capabilities of a population over the generations. The genetic information of each individual is stored in a chromosomal string and the goodness of individual is measured by defining a fitness function based the string. Only the individuals with better characteristics survive during the evolutionary process so that the fitness function is maximized. The detailed procedures of GA is given in Ref. [8].

5.1 The fitness function

In CBR domain, the classification accuracy rate of training case set for a particular weight vector is adopted as the fitness function of GA learning process[9,10]. However, to avoid premature convergence and keep high rate of accuracy, a penalty function is carefully defined based on massive trial analysis. The mathematical form of the fitness and penalty functions designed are expressed as

$$F_l = \sum_{i=1}^{m} [\mathrm{SIM}(T_i, S_k) + P_{li}] \quad \text{with}$$

$p_{li} = 0.5\,\mathrm{SIM}(T_i, S_k)$ if the failure modes of T_i and S_k are same

or

$p_{li} = -1\,\mathrm{SIM}(T_i, S_k)$ if the failure modes of T_i and S_k are different

where F_l is the fitness function of lth weight vector, m is the number of test cases, P_{li} is the penalty function, T_i is the ith test case, and S_k is a reference case which is most similar to T_i, *i. e.*,

$\mathrm{SIM}(T_i, S_k) = \{\max \mathrm{SIM}(T_i, S_j) \mid j = 1, 2, \ldots, n\}$

where S_j is the jth reference case, n is the number of reference cases.

5.2 Training policies

The training of weights for failure attributes has to solve the problem of selection of a test case set and a reference case set. Two kinds of training policies which are normally used in CBR are as follows:

(1) Test_Reference_Set policy

This policy requires the division of the whole failure case set which participates in training into a test set and a reference set. A test ratio can be defined here, which is the proportion of the number of cases in a test set to the number of cases in the whole training set. Given a specified test ratio, an optimal weight vector is searched by GA operation according to the computation of a fitness function described in Section 5.1, by recursively taking each failure case in the test set and matching it with the most similar case in the reference set. During this process, the outcomes (*i. e.* failure modes) of the two matched cases are compared and the success of identification can be judged. Finally, the percentage of the successful matched cases over the whole test case set is counted and signified as the training classification accuracy.

The advantage of applying this training policy is that the optimal weight vector can be searched by means of a limited or relatively small set of testing cases if properly designed. However, different test ratios could give different searched optimal weight vectors. The validation process has to be performed to choose the most robust optimal weight vector that could possess a high popularized capability.

(2) Leave_One_Out policy

This policy takes only one case out of a selected test case set for testing, and matches it with the most similar case in the rest of the case set, and judges the success of match by outcomes of the two cases. And after this, return the case into the test set and take next one. The process repeats until the pre-specified test cycles are satisfied or all cases in the test set are tested. The obvious feature of this policy is the every case in a test set serves as either a test case or a reference case. Therefore, the training classification capability is equally distributed. However, when the test set is large, it will suffer the problem of a computation burden. Despite this, Leave_One_Out policy can still be useful for validation of effectiveness of optimal weights acquired by Policy (1) when it works on a fixed volume of a training set available.

In this study, a mixed training policy is further proposed, which suggests that when the test ratio is relatively small (*e. g.* between 0.25～0.5) Policy (1) is favored; whereas when the test ratio lies between 0.55～1.0, Policy (2) is advocated to work on the test set. The effect of this policy will be demonstrated in Section 6 below.

No matter which training policy is used to acquire the optimal weight vectors, the robustnesses of these weight vectors have to be validated based on the comparison of their training classification accuracies with validated classification accuracies as described in the next subsection. Therefore, certain validation criteria are proposed in this study.

5.3 Validation of training effects

As stated before, finding the most robust optimal weight vector based on the limited test/reference set, which has a good popularized capability, is the aim of the weight learning process. This can be done by the validation of a given optimal weight vector which is acquired by Policy (1) or (2) or a mixed policy through assessment of the weight vector on the whole training set in a case base. In this case, the validated classification accuracy must be recorded by applying this given optimal weight vector on the whole training set by Leave _ One _ Out policy. The validation criteria considered can be described as follows:

(1) Criterion for effectiveness of training

If comparison between a training classification accuracy (denoted by A) for an optimal weight vector and its validated classification accuracy (denoted by B) gives results such that:

(i) A is much higher than B, then the popularized capability of the resulted optimal vector is proved to be poor and ineffective;

(ii) A is much lower than B, then the unreliable or unpredictable results would be anticipated, therefore the resulted weight vector is still regarded as ineffective;

(iii) A is near or equal to B, then the optimal weight vector is considered to be consistent. And furthermore, if A or B is greater than a user specified percentage value (*e. g.*, 70%), the optimal weight vector is proved to be stable and effective.

(2) Criterion for efficiency of training

For an effective optimal weight vector, the comparison of its validated classification accuracy (*i. e.* B value in (iii) above) with the training classification accuracy of directly applying the Leave _ One _ Out policy on the whole training set available is conducted. And if the values of both accuracies are near or same, then the optimal weight vector (which is in fact acquired by applying Policy (1) or (2) or mixed on a limited test case set) is proved to be efficient.

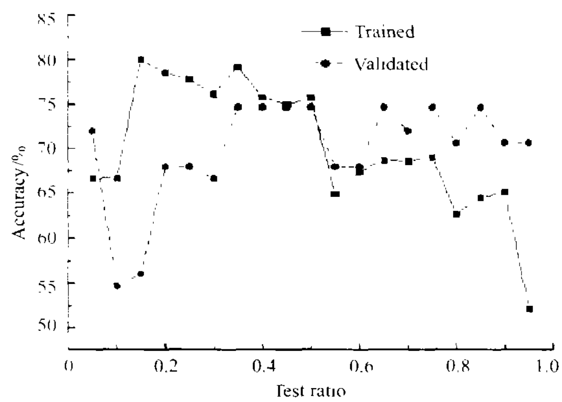
Clearly, only if the effectiveness of training is satisfied, the evaluation of efficiency of training will make sense.

6 Experiments and Discussions

In this study, 358 failure analysis cases of aeronautical equipment have been collected from journals and failure analysis reports. A failure analysis case base is established by using Access database technology. Here selecting a training case subset covering three failure modes for experiments, *i. e.* low _cycle _fatigue _fracture (25), high _cycle _fatigue _fracture (22), and stress _corrosion _inter-granular _fracture (28). The number in brackets denotes the amount of training cases for each failure mode. The reason for such selection is that these modes are in the same level and balanced-distributed. Clearly the training experiment based on only 75 cases is in fact a small sample problem.

6.1 Effects of different training policies

Fig. 3 shows the training and validation results of applying Test _ Reference _ Set policy (*i. e.* Policy (1)) at different test ratios which change from 0.05 to 0.95 by an increment of 0.05. The solid line represents changes of training classification accuracies with test ratios by applying such policy, whereas the dotted line represents changes of validated classification accuracies by applying the optimal weight vector gained through Policy (1) to the whole 75 training case subset.



(Population size: 60 max generations: 300; cross rate: 0.9; mutation rate: 0.05; search interval: [5, 55]; division accuracy: (1)

Fig. 3 Classification accuracy curves by Policy (1)

It can be seen that at interval of 0.05~ 0.35, the training classification accuracies are much higher than the validated classification accuracies. It explains that the sufficient learning samples in a reference set can guarantee the training accuracy, but as a result of too few testing cases in a test set, it may not possess good popularized capabilities, as displayed by low values of validated classification accuracy. At interval of 0.6~ 0.95, the training classification accuracies decreased, which demonstrates learning samples in a reference set is rather limited, so as to unable to give reliable learning classification results. The consistency only exists between test ratios at interval of 0.35～ 0.5, where the training classification accuracies coincide with the validated classification accuracies and keep stable and as high as 74.67%.

Fig.4 showes results by applying the Leave_One_Out policy (i.e. Policy (2)) at different test ratios.

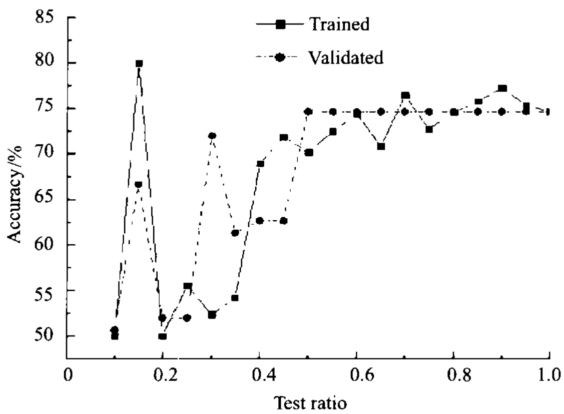


Fig.4 Classification accuracy curves by Policy (2)

It can be seen that when test ratios lies between 0.1～ 0.5, the training classification accuracies and validated classification accuracies oscillate much and show much differences, which proves that the Leave_One_Out policy used for training at small test ratios is ineffective, whereas the results from 0.5～ 1.0 give consistent and convergent values of two classification accuracies, and validates that the classification accuracy keep as high as 74.67% as shown in Fig.3. This demonstrates that when the test case set is relatively large, Policy (2) can safeguard the training effectiveness.

Based on the analysis described above, a mixed training policy is proposed, which suggests that when test ratios lies in 0.25～ 0.5, the Test_Reference_Set policy should be applied, and when the test ratios lies in 0.55～ 1.0 the Leave_One_Out policy should be applieds. Fig.5 gives results of applying the mixed training policy.

Clearly, both curves of training classification accuracy and validated classification accuracy show much improved consistency, and the validated classification accuracy of 74.67% maintains constant since the test ratio of 0.35. According to validation

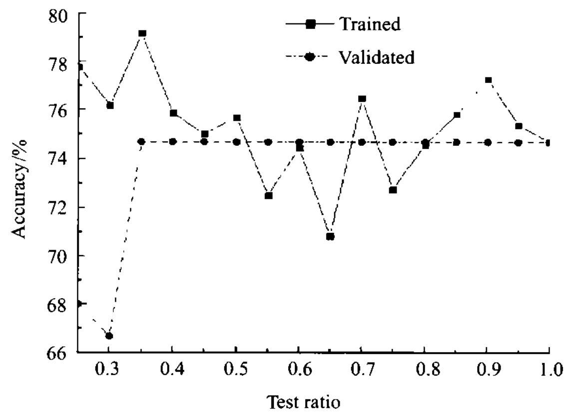


Fig.5 Classification accuracies by the mixed policy

criteria described in Subsection 5.3, the mixed policy gives better training effects than the other two training policies and possesses the highly efficient optimal weight vectors. Table 1 lists the best equivalent weight vectors at different test ratios in applying the mixed training policy.

Table 1 Best weight vectors at different test ratios

Test ratios	Best equivalent weight vectors
0.35, 0.4, 0.45, 0.5, 0.7, 0.75, 0.8, 0.85	(2, 1, 1)
0.55, 0.6, 0.65, 0.9, 0.95	(3, 1, 1)
1.0	(5, 1, 1)

The robust optimal weight vector should be (2, 1, 1) which occurs most frequently among all the test ratios.

6.2 Test on identification capability using other failure modes

To check the popularized capability of the optimal weight vector gained through above learning process, the application of the weight vector (2, 1, 1) to identification of other two failure modes by CBR process is conducted. In this case, 15 brittle_cleavage_fracture failure mode cases and 14 thermal_fatigue_fracture failure mode cases are considered. Table 2 shows the identification results and their comparison with the training effects by applying Leave_One_Out policy alone to this 28 failure cases.

It can be seen from Table 2 that the percentage values of identification by (2, 1, 1) vector and by Leave_One_Out policy are quite near and high, which proves that the optimal weight vector

(2, 1, 1) obtained through above 75 training cases can be popularized to other failure mode identification cases.

Table 2 The recognition effect on other patterns by weight vector (2, 1, 1)

Failure modes	No. of cases	Identification by (2, 1, 1)	Leave_One_Out effects
Brittle_cleavage_fracture	15	11(73.3%)	11(73.3%)
Thermal_fatigue_fracture	14	12(85.7%)	13(92.8%)

6.3 Training effects of mixed unbalanced-distributed failure mode cases

Fig.6 shows the training results of mixing 75 Failure cases described in Subsection 6.1 with 28 Failure cases described in Subsection 6.2 The total of 5 failure modes are considered which are clearly unbalanced-distributed. The training policy adopted is the mixed policy described in Subsection 6.1.

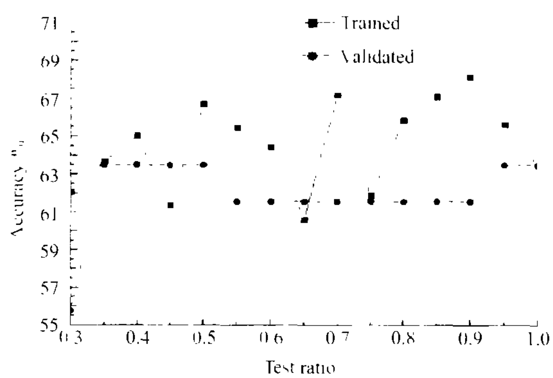


Fig.6 Classification accuracies for 5 modes

The training curves show that the stable validated classification accuracy reached 63.46% when the test ratios are at intervals of 0.35~ 0.5 and 0.95~ 1.0. Another stable value is 61.46% which is slightly less at interval of 0.55~ 0.9. This demonstrates that the mixed training policy can still achieved relatively better results of identification for unbalanced-distributed failure case set. Nevertheless, it is argued that the best training effect would be taken in the most balanced-distributed failure mode cases. Table 3 lists the optimal weight vectors for the 5 failure mode identification processes with different test ratios.

Again, it is shown that the robust optimal weight vector is (2, 1, 1) as acquired in Subsection 6.2.

Table 3 Test ratios *vs* the best weight vectors

Test ratios	Best equivalent weight vectors
0.35, 0.4, 0.45, 0.5, 0.95, 1.0	(2, 1, 1)
Others	(1, 1, 1)

7 Conclusions

The GA based weight training process has been conducted in a CBR system for failure mode identification of aeronautical equipment. Failure attributes used for matching in case retrieval phase are grouped into three categories, and a weighed K Nearest Neighbor approach is adopted for similarity measures between an old or stored case and the new or target case. The fitness function of GA for evolutionary searching of optimal weight vectors is carefully studied. The performance of the system is tested based on the consideration of three kinds of training policies, in which the mixed training policy is newly proposed by the combined use of Test_Reference_Set policy and Leave_One_Out policy at different test ratios. The evaluation or validation of training effects are studied and novel validation criteria are set up.

The experimental results show that

(1) The sufficient and balanced-distributed failure mode training case set can give better training results than unbalanced ones.

(2) The highly efficient optimal weight vector can be achieved by using the mixed training policy.

(3) The popularized capability of an optimal weight vector by applying the mixed training policy is effective and robust for identification of more failure modes, and is valid for practical use.

References

[1] 钟群鹏, 张峥, 田永江. 机械装备失效分析诊断技术[J]. 北京航空航天大学学报, 2002, 28(5): 497- 502.
Zhong Q P, Zhang Z, Tian Y J. Failure analysis diagnosis of mechanical equipment[J]. Journal of Beijing University of Aeronautics and Astronautics, 2002, 28(5): 497- 502. (in Chinese)

[2] Maye D. Boiler tube failure mechanism recognition—an expert system[J]. CIM Bulletin, 1990, 83(939): 92- 95.

[3] Komai K, Minoshima K I. Recognition of different fracture

surface morphologies using computer image processing technique[J]. JSME Int J Series A, 1993, 36(2): 220- 227.
[4] Liao T W, Zhan Z H, Mount C R. An intergrated database and expert system for failure mechanism identification: Part I and Part II [J]. Engineering Failure Analysis, 1999(6): 387- 421.
[5] Liao T W, Zhan Z H, Mount C R. A case based reasoning system for identifying failure mechanisms[J]. Artificial Intelligence, 2000 (13): 199- 213.
[6] 徐元铭, 骆晶妍, 邢小楠. 航空机械装备失效分析专家系统应用研究[J]. 机械设计与研究, 2003, 19(1): 53- 56.
Xu Y M, Luo J Y, Xing X N. The application of expert system for failure analysis of aeronautical equipment[J]. Machine Design and Research, 2003, 19(1): 53- 56. (in Chinese)
[7] 徐元铭, 张洋. 实例推理技术在航空动部件失效分析中的应用[J]. 材料工程, 2003(增刊): 289- 291.
Xu Y M, Zhang Y. The application study of case based reasoning for failure analysis of aeronautical equipment[J]. Material Engineering, 2003(Sup): 289- 291. (in Chinese)
[8] Goldberg D E. Genetic algorithms in search, optimization and machine learning [M]. London: Addision-Wesley; 1989.
[9] Shin K S, Han I G. Case based reasoning supported by genetic algorithms for corporate bond rating[J]. Expert System with Applications, 1999(16): 85- 95.
[10] 王玉, 邢渊, 阮雪榆. 基于事例的推理循环中人工神经网络和遗传算法的4种应用模型[J]. 上海交通大学学报, 2003, 37(2): 202- 204.
Wang Y, Zhou Y, Ruan X Y. Four application models for artificial neural network and genetic algorithm in case based reasoning cycle[J]. Journal of Shanghai Jiaotong University, 2003, 37(2): 202- 204. (in Chinese)

Biographies:

XU Yuan-ming (1965 -) He received his Ph. D degree in Aeronautical Engineering. He is now a professor and doctorial student supervisor of Beijing University of Aeronautics and Astronautics. His research interests include: aircraft structural optimization, intelligent CAD, intelligent based fault diagnosis and failure analysis.